mentations, multiple capture devices can be available to the sender 3D conversation system, and which devices are used to capture data can be based on a conversation context, such as available bandwidth, a configuration of a receiving 3D conversation system, a viewpoint of a receiving user, etc. The captured data can be initially tagged with meta-data such as time of capture and with an identifier of the device that captured it. Additional capture stage details are discussed below in relation to block **436** of FIG. **4**, block **504** of FIG. **5**, and FIG. **8**.

[0024] The tag and filter stage can include various processes to tag the captured data with further meta-data, improve the quality of captured data, and/or remove unnecessary portions of the captured data. The captured data can be tagged with calibration data generated at the calibration stage (discussed below) signifying intrinsic and extrinsic parameters (e.g., a camera position and orientation, camera geometries, etc.), objects or people identified in sequences of the images, the areas of images showing identified objects or people, results of analysis (e.g., adding a user skeleton view), video motion characteristics, etc. Various captured data streams can also be cross-augmented by using multiple related video streams to enhance each other. For example, where color images and depth images are taken from similar positions, the color image data and depth image data can be cross-applied between the data streams to enhance each other. Finally, portions of the images can be removed, such as the background of the user. Additional tag and filter stage details are discussed below in relation to block **438** of FIG. **4**, block **506** of FIG. **5**, and FIG. **9**.

[0025] The compression stage can transform the captured data into a format for transmission across a network (e.g., by applying a video codec or other compression algorithm) and the decompression stage can transform the compressed data back to a version (e.g., via lossy or lossless compression) of the original data (e.g., back into individual images or videos, point clouds, light fields, etc.). In various implementations, the meta-data tagged to the various data streams can be encoded into the compressed video stream or can be provided as separate associated data. Additional compression stage and decompression stage details are discussed below in relation to blocks **440** and **442** of FIG. **4**, blocks **508** and **510** of FIG. **5**, and FIGS. **10** and **11**.

[0026] The reconstruction stage can create a 3D representation of the sending user. The reconstruction stage can perform this transformation of the captured depth data into a 3D representation such as a point cloud, a signed distance function, populated voxels, a mesh, a light field, etc., using the calibration data to combine data from multiple sources and/or transform the captured data into position and contour information in 3D space. For example, each pixel in a depth image depicting a user can be transformed into a 3D representation of at least part of the user by applying transformations based on the intrinsic and extrinsic properties of the camera. The transformations can take each pixel taken at the camera location and determine a corresponding point in 3D space representing a point on the surface of the user. In some implementations, the reconstruction stage can also apply shading or color data to the 3D representation based on the calibration data. In some cases, the reconstruction process can be customized based on the computational and display characteristics of the receiving 3D conversation system. In some implementations, the 3D representation can include portions that are not direct translations of captured data, e.g., for portions of the user that were not depicted in the captured data. These portions can be e.g., avatar representations, machine learning estimations of the missing portions, or previously captured versions of the missing portions. Additional reconstruction stage details are discussed below in relation to block **444** of FIG. **4**, block **512** of FIG. **5**, and FIG. **12**.

[0027] The render stage can generate one or more 2D images from a viewpoint of the receiving user based on the 3D representation generated by the reconstruction stage. While displayed to the receiving user as 2D images (unless the receiving user has a true 3D display), these can appear to the receiving user to be a 3D representation of the sending user. These images can be generated to meet the display properties of the receiving system, e.g., to match resolution, display size, or display type of the receiving system. For example, where the receiving system is an artificial reality system with a display for each eye, the render stage can generate an image from the viewpoint of each eye at the resolution of these displays. In various implementations, the render stage can generate a single image, two "stereo" images, a light field, etc. In some implementations, the render stage can transform captured color data and apply it to the rendered images. Additional render stage details are discussed below in relation to block **446** of FIG. **4**, block **514** of FIG. **5**, and FIG. **13**.

[0028] The display stage can receive the rendered one or more 2D images and output them via display hardware of the receiving system. For example, the display stage can display the image(s) on a screen, project them onto a "virtual cave" wall, project them into a user's eye, etc. The display stage can also synchronize display of the 2D images with output of corresponding audio. Additional display stage details are discussed below in relation to block **448** of FIG. **4**, block **516** of FIG. **5**, and FIG. **14**.

[0029] An additional calibration stage can be also be included in the pipeline which, in various implementations, can be performed as a pre-stage to the 3D conversation (e.g., an automatic or manual process partially or completely performed by a system administrator, manufacturer, or a user) and/or can be performed "online" as the 3D conversation takes place. The calibration stage can gather intrinsic and extrinsic properties of cameras that are part of a sending system. Intrinsic parameters can specify features of a camera that are internal (and often generally fixed) for a particular camera. Examples of intrinsic parameters include focal length, a relationship between a pixel coordinates, lens geometric distortion, etc. These parameters can characterize the optical, geometric, and digital characteristics of the camera, allowing a mapping between camera coordinates and pixel coordinates of an image. Extrinsic parameters can specify conditions or context external to the camera. Examples of extrinsic parameters include the location and orientation of the camera, ambient conditions (e.g., heat, moisture, etc.), lighting characteristics (e.g., lighting source location, type, orientation), etc. These parameters can be used to characterize light received at each camera pixel, allowing that light to be interpreted in terms of a 3D environment. Each camera used by the 3D conversation system can be individually calibrated and associated with resulting calibration meta-data. Additional calibration stage details are discussed below in relation to block **434** of FIG. **4**, block **502** of FIG. **5**, and FIG. **7**.